Brute Force vs. Simulated Annealing For SRAM Optimization

Contents

Introduction

ED Curves for 8k and 16k SRAM in Fig1a & Fig2a were generated from Vipro, simulated annealing is then applied to find each optimum point in the curve, both using constraint search and pareto-optimum.

The number of iterations consumed by brute force and simulated annealing are compared using the bar graph below in Fig1b, Fig1c, Fig2b and Fig2b.

8K SRAM

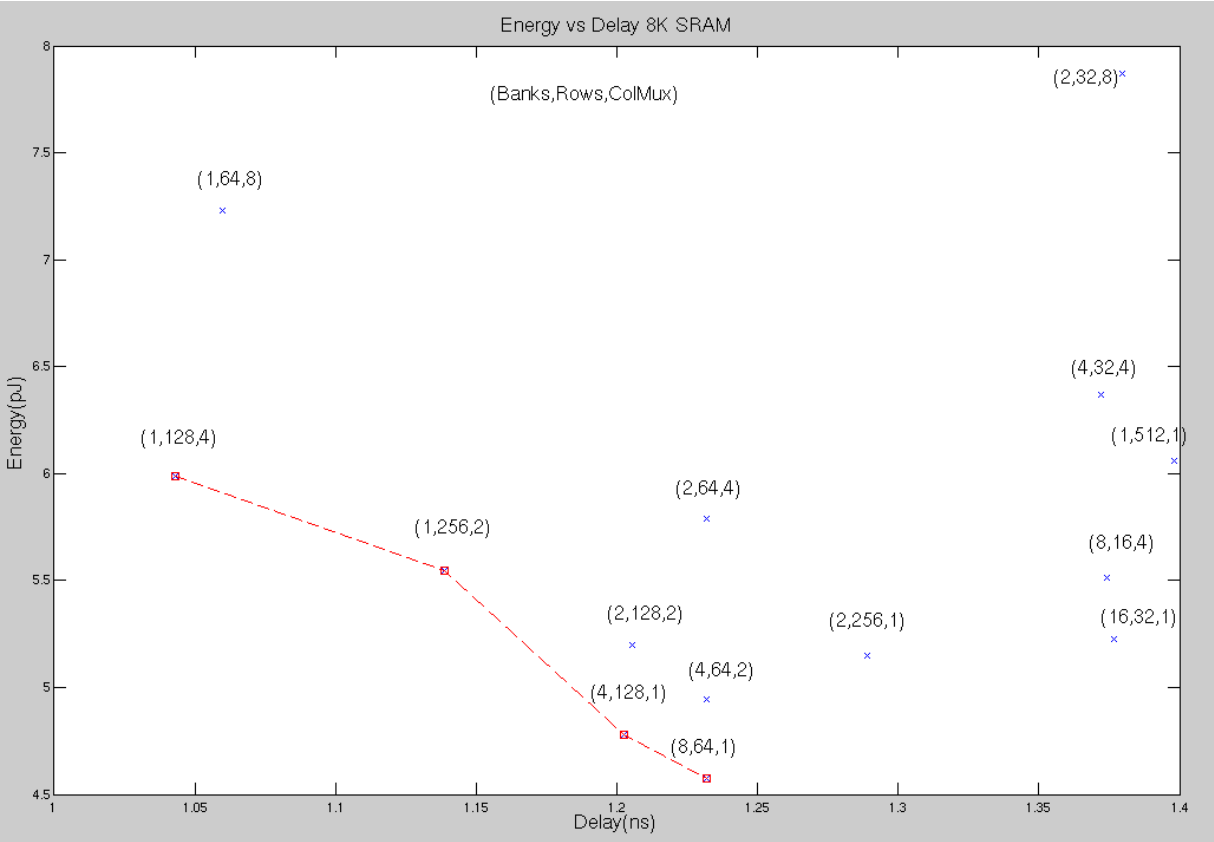

Fig1a. 8K SRAM, ED Curve

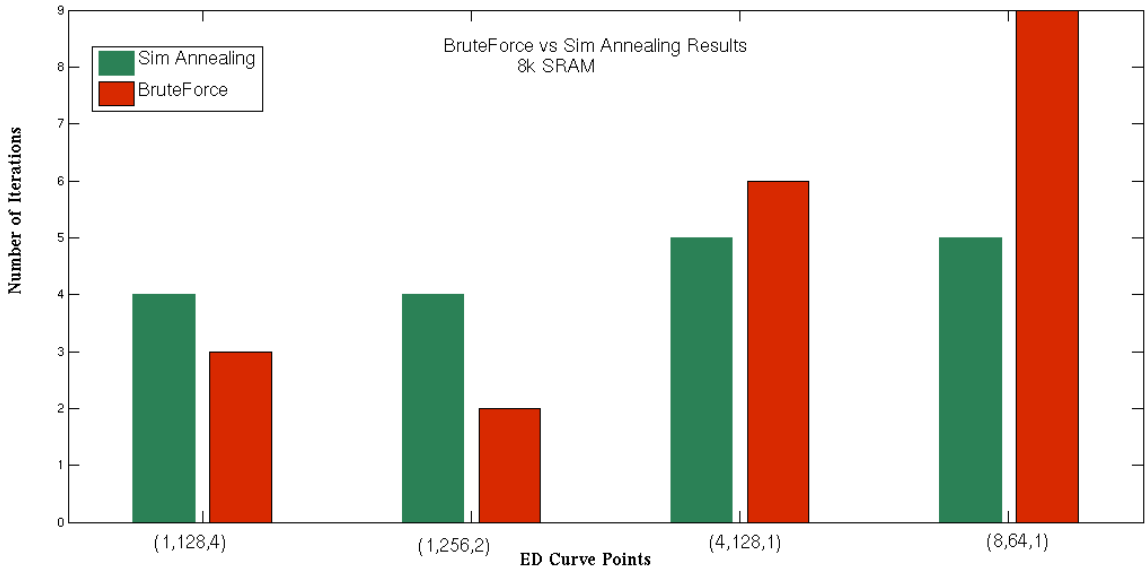

Fig1b. 8K SRAM, Constraint Points

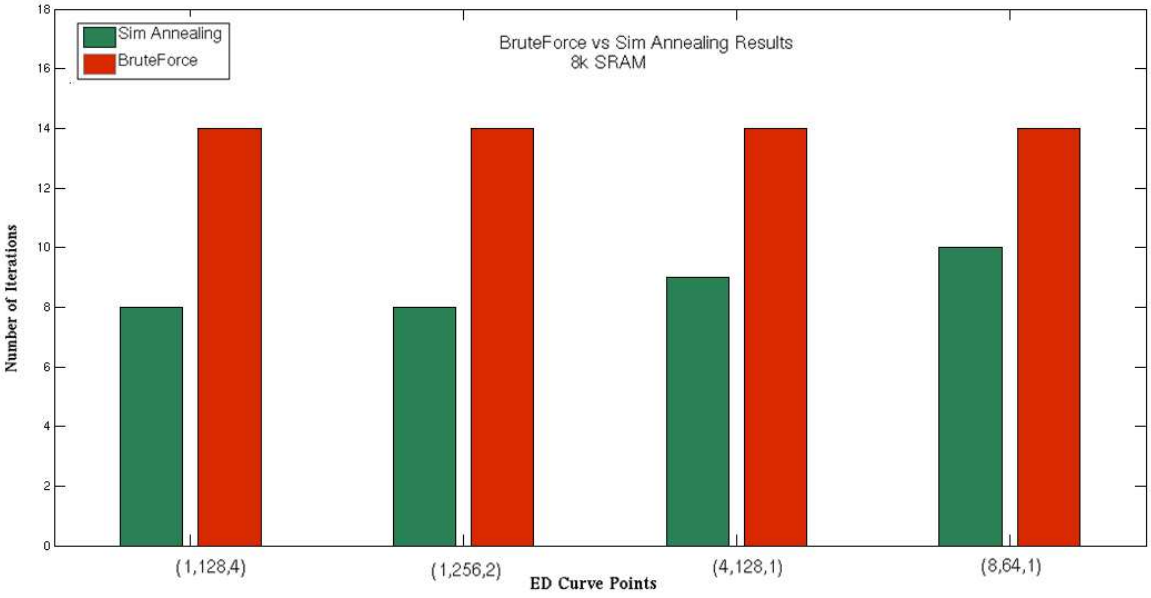

Fig1c. 8K SRAM, Pareto Optimum Points

16K SRAM

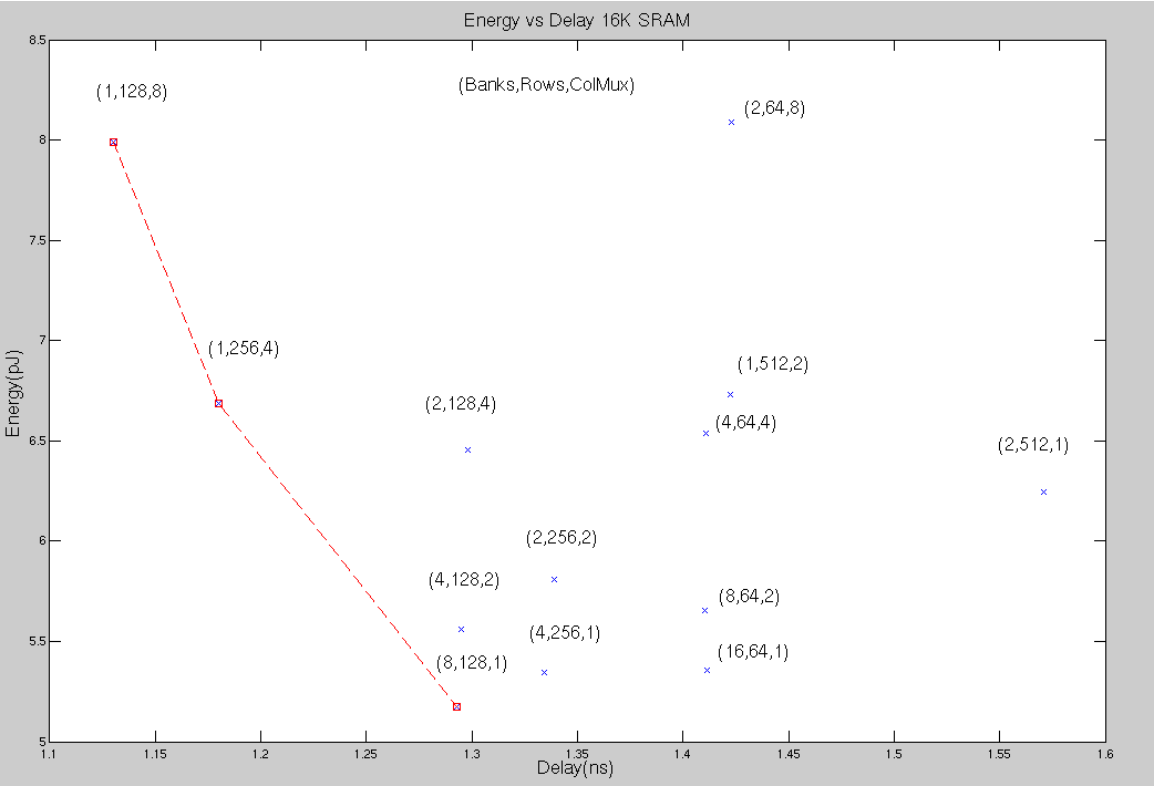

Fig2a. 16K SRAM, ED Curve

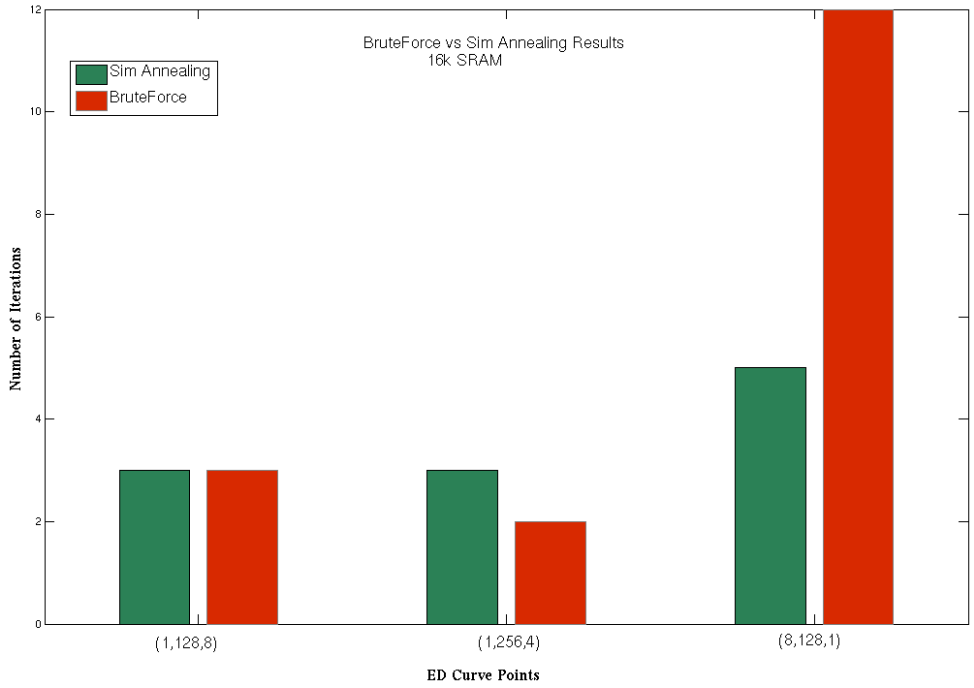

Fig2b. 16K SRAM, Constraint Points

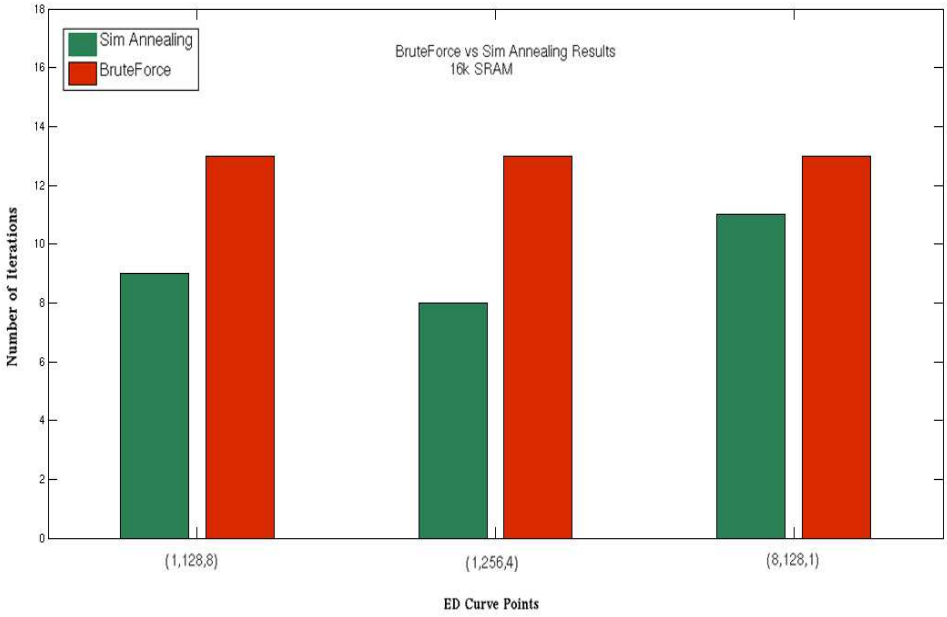

Fig2c. 16K SRAM, Pareto Optimum Points

Conclusion

According to the results shown from optimization, the efficiency of the simulated annealing optimization approach increases by the increase in the design space and/or the number of optimization knobs.

In small designs like the one presented here it is still useful to apply optimization over brute force.

From the 8K & 16K, Pareto-optimum curves at Fig1c & Fig2c simulated annealing outperformed BruteForce.

In the 8K & 16K, constraint point cases the optimum points were the first points brute force scanned. That makes brute force performance comparable to the optimization results. However if the order of brute force scan was reversed or the optimum points were located at the end of the search then brute force would be much expensive even in small design.

On the reverse, simulated annealing doesn't suffer from this ordering problem as it can be independent from the starting point through adapting its parameters (temperature) to find an optimum point at first then do fine tuning/search.

In all of the simulation results above, simulated annealing started with the same configuration as brute force started with, the minimum number of banks and column mux. In 8K SRAM, it started with (1,512,1) and in 16K SRAM it started with (1,512,2), where (b,r,c) indicates b=number of banks, r=number of rows and c=number of column mux.